



DVaaS Detective: The Case of the Failing Tests

Victor Rios (presenter), Steffen Smolka

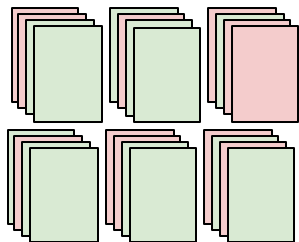


P4-BAR



Debugging Pain

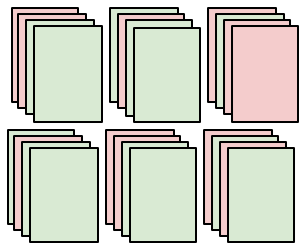
Test Outcomes



Debugging Pain

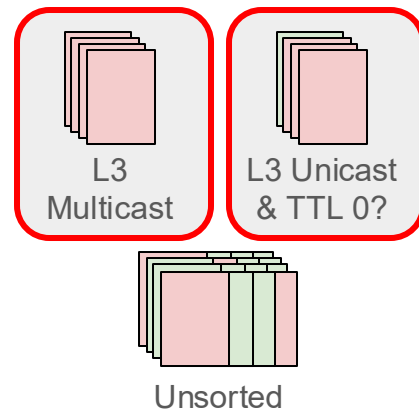
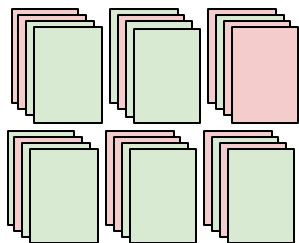
BEFORE:

Test Outcomes



Debugging Pain

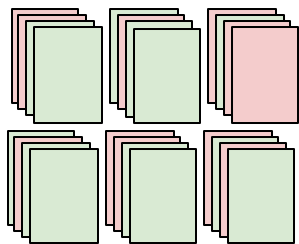
BEFORE:
Test Outcomes



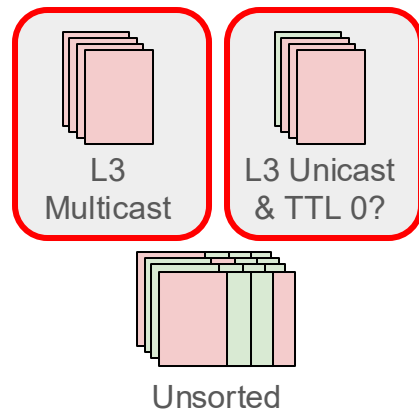
Debugging Yay

BEFORE:

Test Outcomes



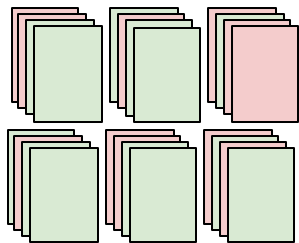
AFTER:



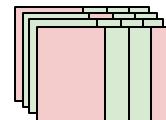
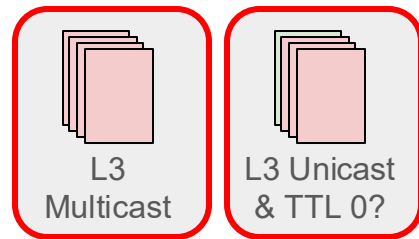
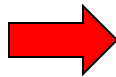
Debugging Yay

BEFORE:

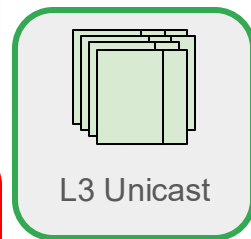
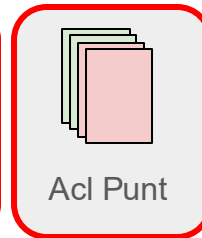
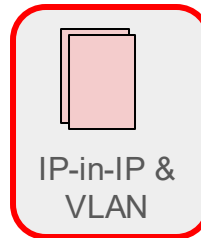
Test Outcomes



AFTER:



Unsorted



Agenda

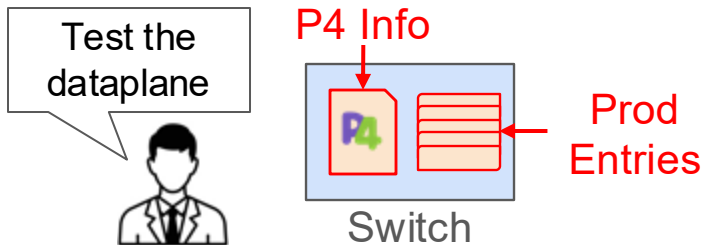
Debugging Pain

Background and Problem with DVaaS

Solution: DVaaS Detective

Status & Future Work

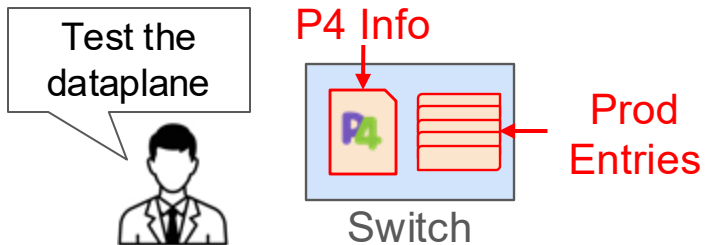
Dataplane Validation as a Service (DVaaS)



DVaaS



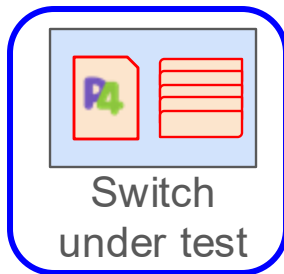
Dataplane Validation as a Service (DVaaS)



DVaaS



This is
the input!



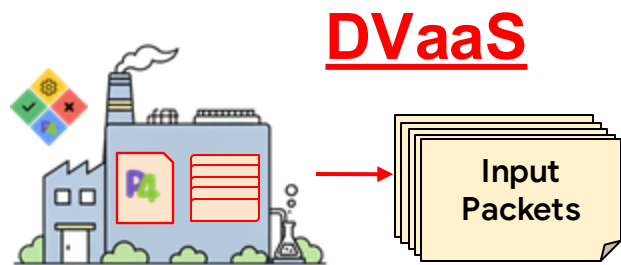
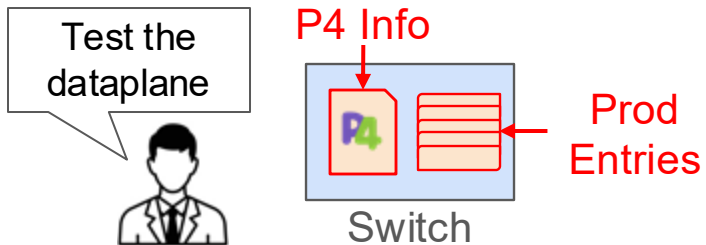
Test Outcomes



Summary

This is the
output!

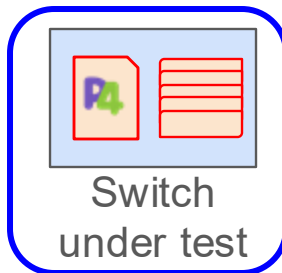
Dataplane Validation as a Service (DVaaS)



P4-Symbolic
Packet
Synthesizer



**This is
the input!**



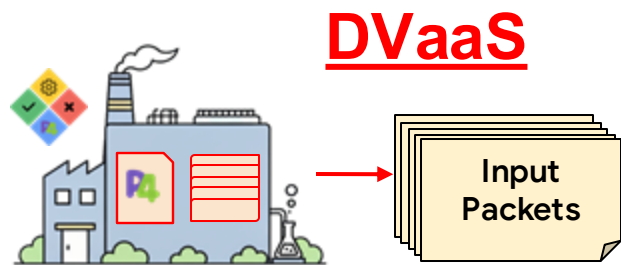
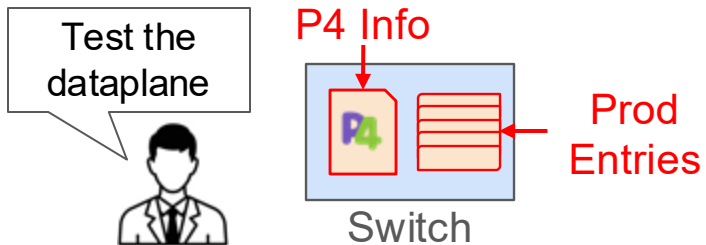
Test Outcomes



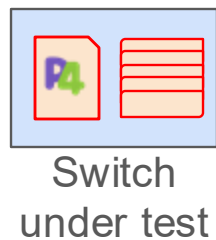
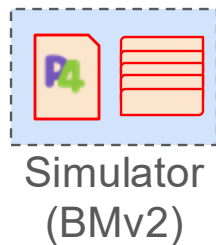
Summary

**This is the
output!**

Dataplane Validation as a Service (DVaaS)



This is the input!



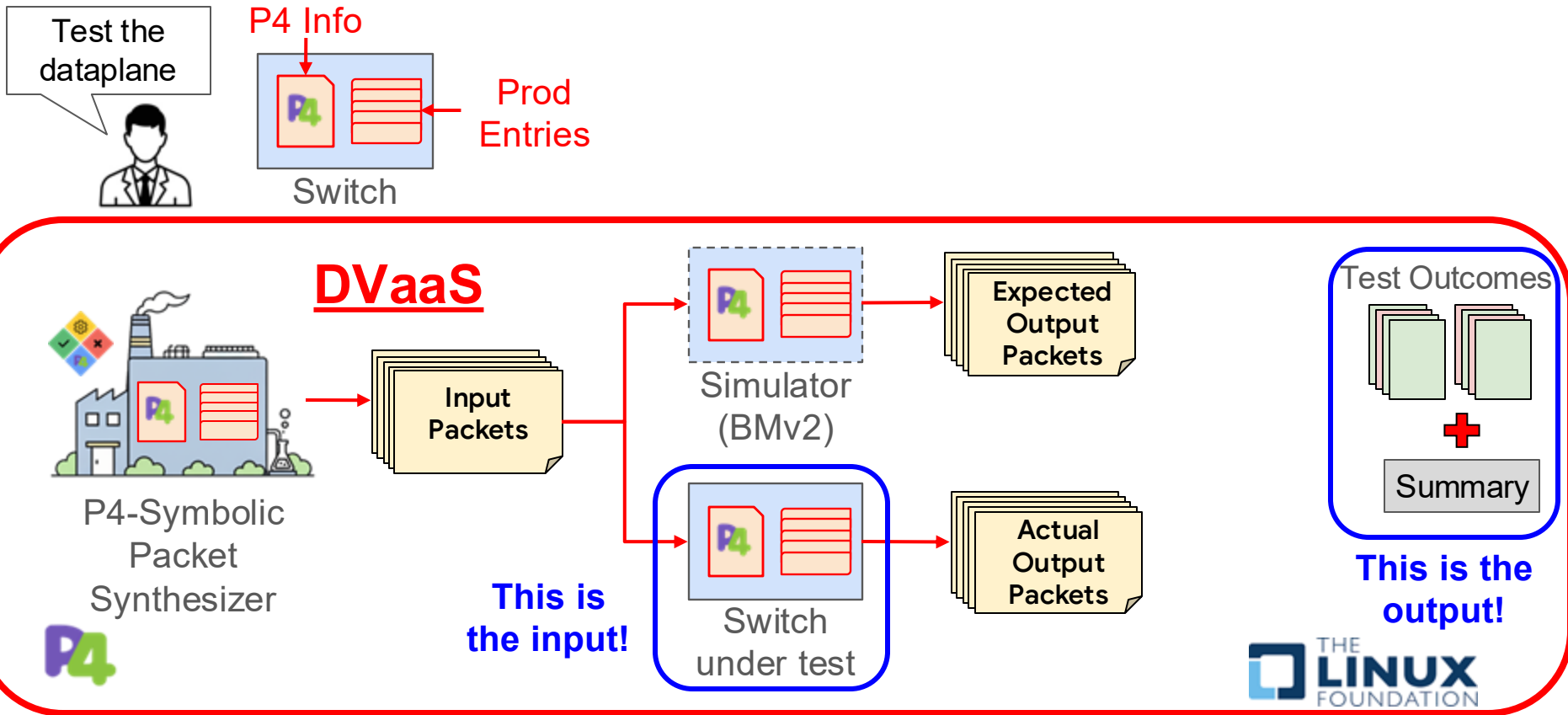
Test Outcomes



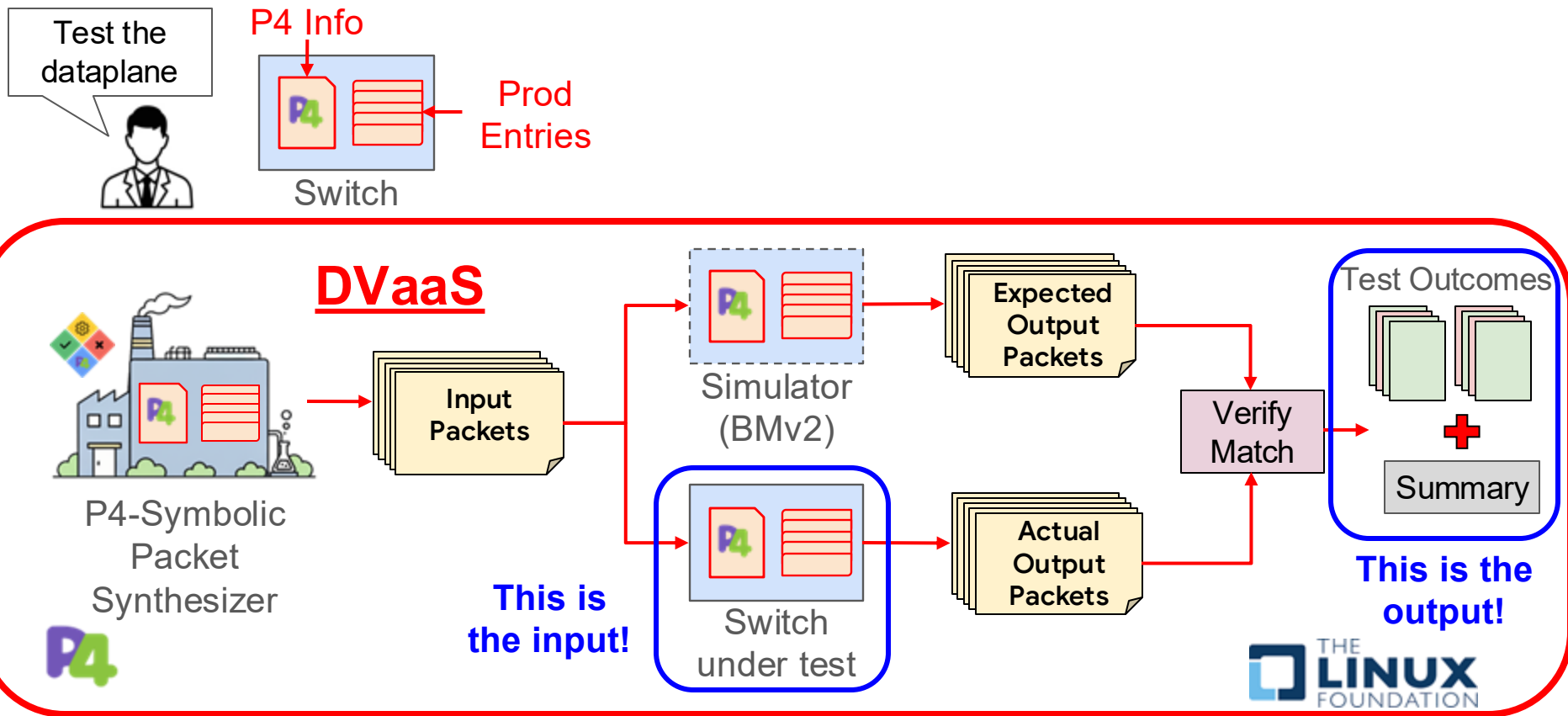
Summary

This is the output!

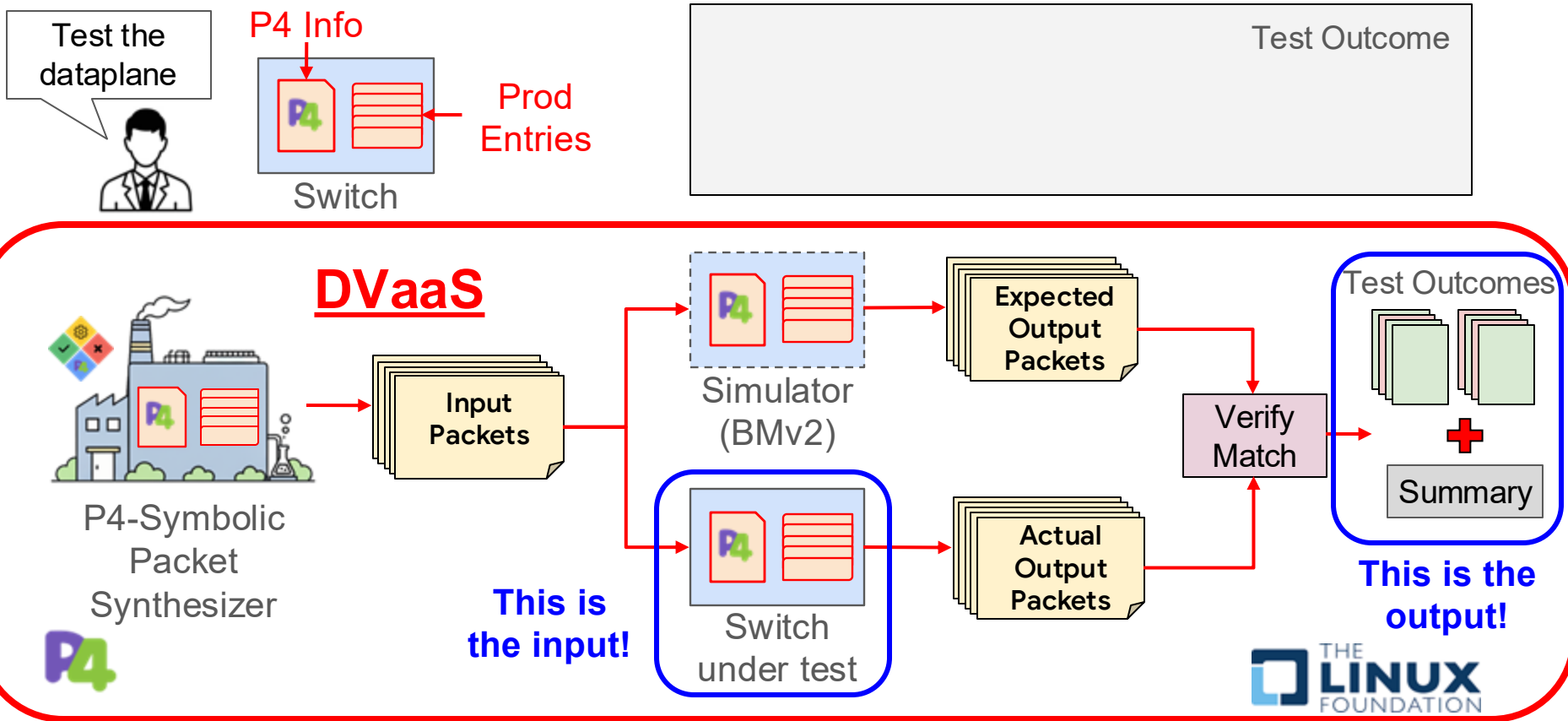
Dataplane Validation as a Service (DVaaS)



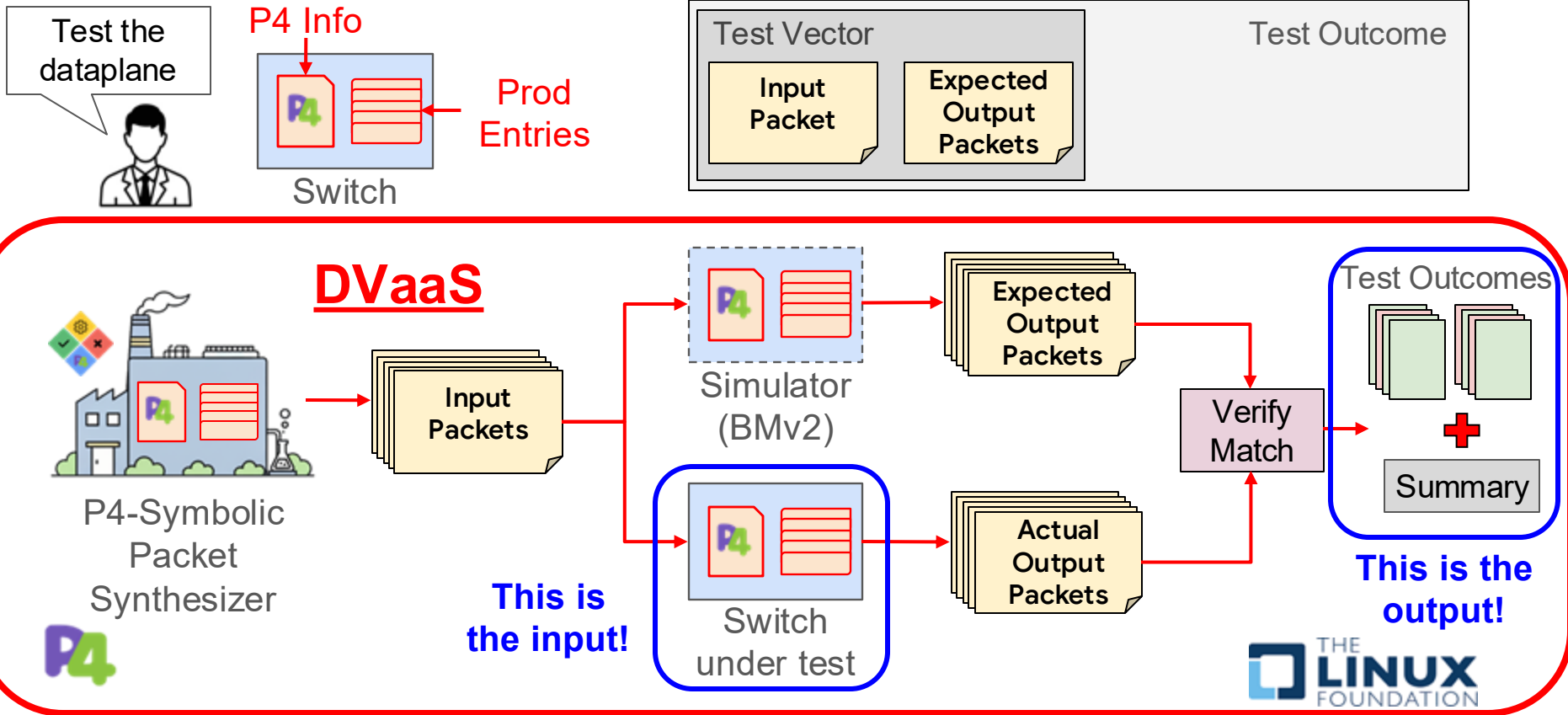
Dataplane Validation as a Service (DVaaS)



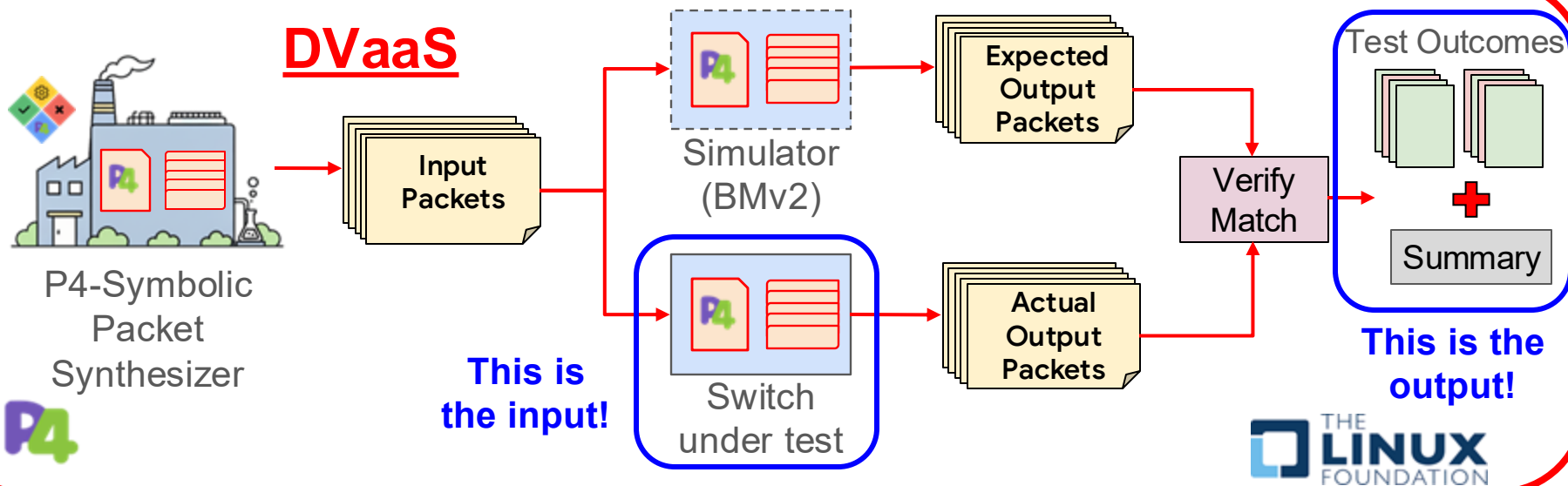
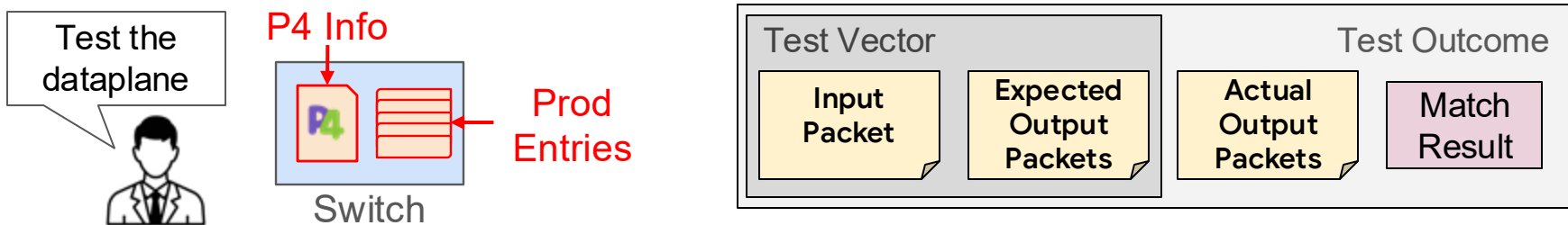
Dataplane Validation as a Service (DVaaS)



Dataplane Validation as a Service (DVaaS)



Dataplane Validation as a Service (DVaaS)



The Problem With DVaaS

- DVaaS excels at finding issues

We found
an issue



Test Outcomes



Summary

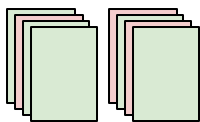
The Problem With DVaaS

- DVaaS excels at finding issues
- Not so much at explaining issues

What feature
is broken?!



Test Outcomes



Summary

The Problem With DVaaS

- DVaaS excels at finding issues
- Not so much at explaining issues
- Tons of micro-level details

What feature
is broken?!



Test Outcomes



Summary

Test Outcome

```
== RESULT =====
Expected: DATAPLANE packet gets forwarded (1 copies)
Actual: DATAPLANE packet got dropped
== INPUT =====
type: DATAPLANE
packet {
  port: "1"
  headers {
    ethernet_header {
      ethernet_destination: "ff:ee:dd:cc:bb:aa"
    }
  }
}
...
== EXPECTED OUTPUT =====
packets {
  port: "8"
  headers {
    ethernet_header {
      ethernet_destination: "06:05:04:03:02:01"
    }
  }
}
...
== ACTUAL OUTPUT =====
<drop>
== EXPECTED INPUT-OUTPUT TRACE (P4 SIMULATION) =====
Table 'acl_table': miss
Table 'ipv4_route_table': hit
Match: ipv4_dst: 10.0.0.0/8
Action: forward(port: 42)
Primitive: 'mark_to_drop' (routing.p4(275))
...
```

The Problem With DVaaS

- DVaaS excels at finding issues
- Not so much at explaining issues
- Tons of micro-level details...

tons...

What feature
is broken?!



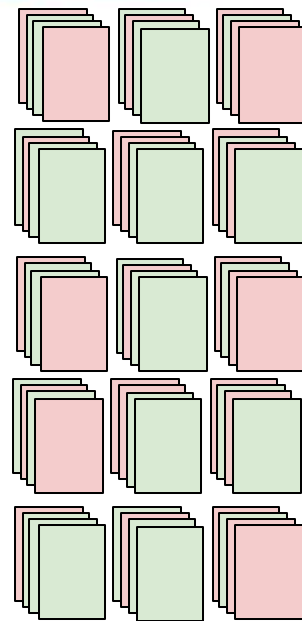
Test Outcomes



Summary

Test Outcome...s

```
== RESULT =====
Expected: DATAPLANE packet gets forwarded (1 copies)
Actual: DATAPLANE packet got dropped
== INPUT =====
type: DATAPLANE
packet {
  port: "1"
  headers {
    ethernet_header {
      ethernet_destination: "ff:ee:dd:cc:bb:aa"
    }
  }
}
...
== EXPECTED OUTPUT =====
packets {
  port: "8"
  headers {
    ethernet_header {
      ethernet_destination: "06:05:04:03:02:01"
    }
  }
}
...
== ACTUAL OUTPUT =====
<drop>
== EXPECTED INPUT-OUTPUT TRACE (P4 SIMULATION) =====
Table 'acl_table': miss
Table 'ipv4_route_table': hit
Match: ipv4_dst: 10.0.0.0/8
Action: forward(port: 42)
Primitive: 'mark_to_drop' (routing.p4(275))
...
```



The Problem With DVaaS

- DVaaS excels at finding issues
- Not so much at explaining issues
- Tons of micro-level details...
tons...

What feature
is broken?!



Test Outcomes



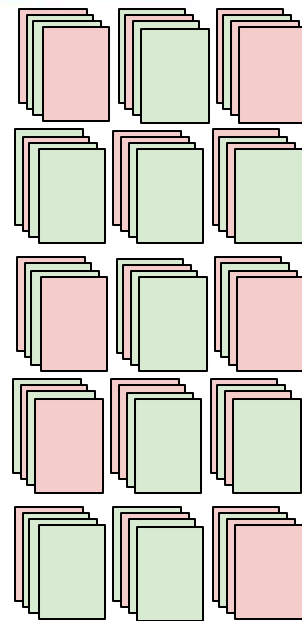
Summary



This is not enough!

Test Outcome...s

```
== RESULT =====
Expected: DATAPLANE packet gets forwarded (1 copies)
Actual: DATAPLANE packet got dropped
== INPUT =====
type: DATAPLANE
packet {
  port: "1"
  headers {
    ethernet_header {
      ethernet_destination: "ff:ee:dd:cc:bb:aa"
    }
  }
}
...
== EXPECTED OUTPUT =====
packets {
  port: "8"
  headers {
    ethernet_header {
      ethernet_destination: "06:05:04:03:02:01"
    }
  }
}
...
== ACTUAL OUTPUT =====
<drop>
== EXPECTED INPUT-OUTPUT TRACE (P4 SIMULATION) =====
Table 'acl_table': miss
Table 'ipv4_route_table': hit
Match: ipv4_dst: 10.0.0.0/8
Action: forward(port: 42)
Primitive: 'mark_to_drop' (routing.p4(275))
...
```



Summary

3943 (98.55%) of 4001 test vectors passed

The System Under Test (SUT):

- forwarded: for 1902 test vectors (expected: 1960)
- punted: for 1974 test vectors (expected: 1974)
- dropped: for 741 test vectors (expected: 683)

The Problem With DVaaS

- DVaaS excels at finding issues
- Not so much at explaining issues
- Tons of micro-level details...

tons...

What feature
is broken?!



Test Outcomes
macro-level analysis



Summary

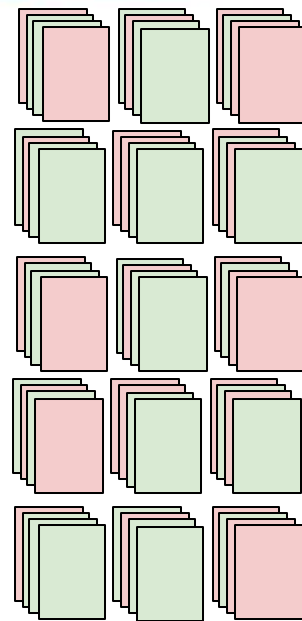
Explaining results
is labor-intensive!



This is not enough!

Test Outcome...

```
== RESULT =====  
Expected: DATAPLANE packet gets forwarded (1 copies)  
Actual: DATAPLANE packet got dropped  
== INPUT =====  
type: DATAPLANE  
packet {  
  port: "1"  
  headers {  
    ethernet_header {  
      ethernet_destination: "ff:ee:dd:cc:bb:aa"  
    }  
  }  
}  
...  
== EXPECTED OUTPUT =====  
packets {  
  port: "8"  
  headers {  
    ethernet_header {  
      ethernet_destination: "06:05:04:03:02:01"  
    }  
  }  
}  
...  
== ACTUAL OUTPUT =====  
<drop>  
== EXPECTED INPUT-OUTPUT TRACE (P4 SIMULATION) =====  
Table 'acl_table': miss  
Table 'ipv4_route_table': hit  
Match: ipv4_dst: 10.0.0.0/8  
Action: forward(port: 42)  
Primitive: 'mark_to_drop' (routing.p4(275))  
...
```



Summary

3943 (98.55%) of 4001 test vectors passed

The System Under Test (SUT):

- forwarded: for 1902 test vectors (expected: 1960)
- punted: for 1974 test vectors (expected: 1974)
- dropped: for 741 test vectors (expected: 683)



Agenda

Debugging Pain

Background and Problem with DVaaS

Solution: DVaaS Detective

Status & Future Work

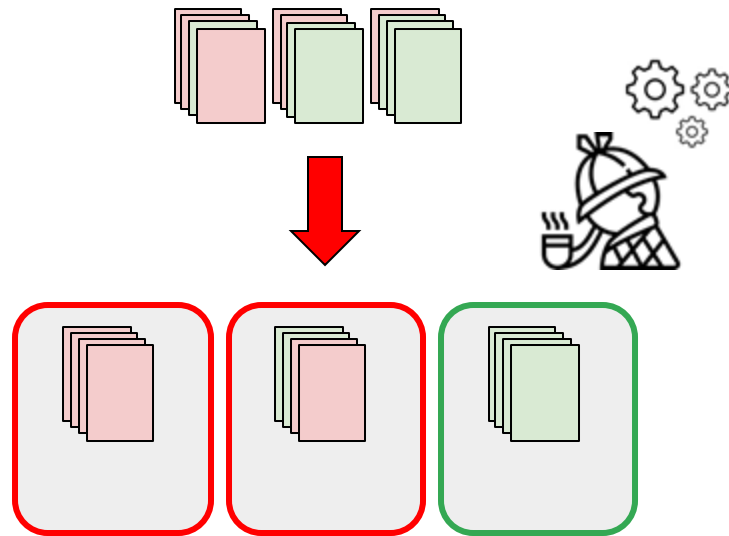
The Solution: DVaaS Detective

- Macro-level analysis on test outcomes



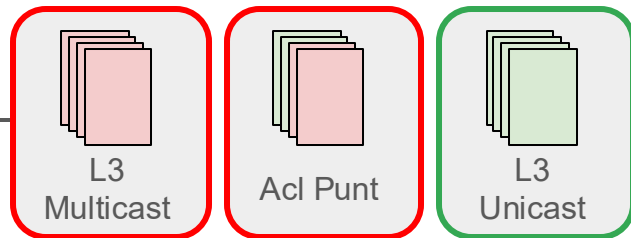
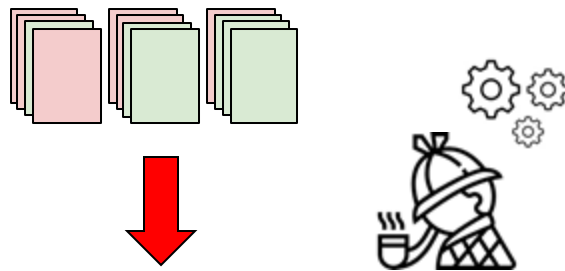
The Solution: DVaaS Detective

- Macro-level analysis on test outcomes
- Classifies test outcomes



The Solution: DVaaS Detective

- Macro-level analysis on test outcomes
- Classifies test outcomes
- Generates concise explanations



DVaaS Detective: Found the following pattern(s) that match FAILING TEST VECTORS:

```
- IF # expected output packets >= 3  
  && admit to l3 table hit == TRUE
```

**What kind of
test vectors?**

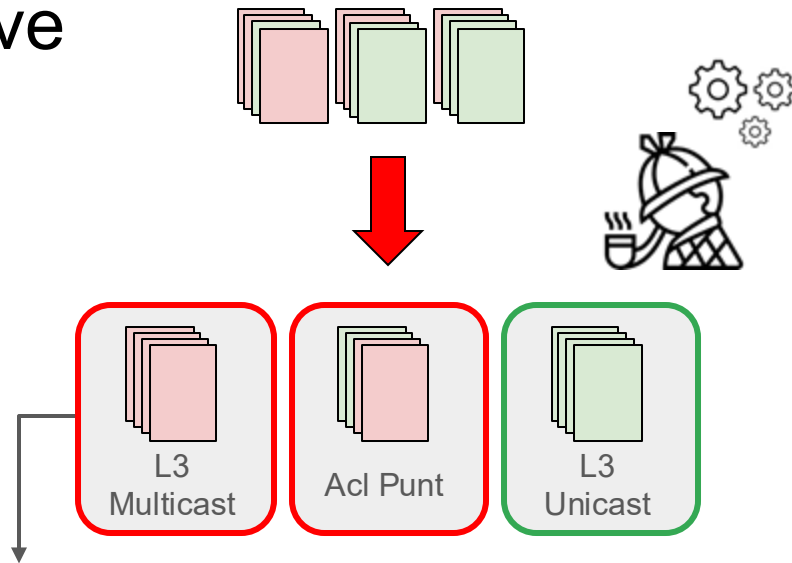
THEN test vector FAILS

* coverage: 67%, accounting for 100 out of 150 failing test vectors

* accuracy: 100%, 150 out of 150 test vectors that match the conditions fail

The Solution: DVaaS Detective

- Macro-level analysis on test outcomes
- Classifies test outcomes
- Generates concise explanations
- Insight into broken/working features



How many failures are explained by this?

Are there exceptions to this rule?

DVaaS Detective: Found the following pattern(s) that match FAILING TEST VECTORS:

```
- IF # expected output packets >= 3  
&& admit to l3 table hit == TRUE
```

```
THEN test vector FAILS
```

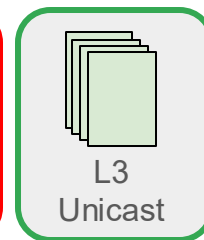
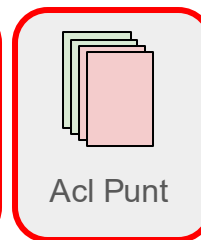
```
* coverage: 67%, accounting for 100 out of 150 failing test vectors
```

```
* accuracy: 100%, 150 out of 150 test vectors that match the conditions fail
```

What kind of test vectors?

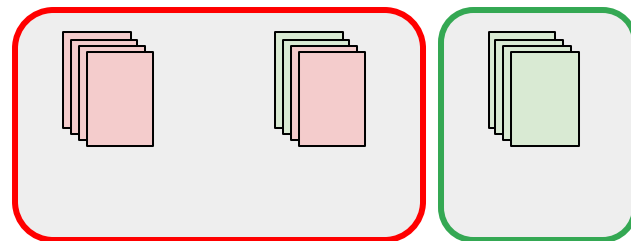
Problem Analysis

Test Outcomes



Problem Analysis

Test Outcomes

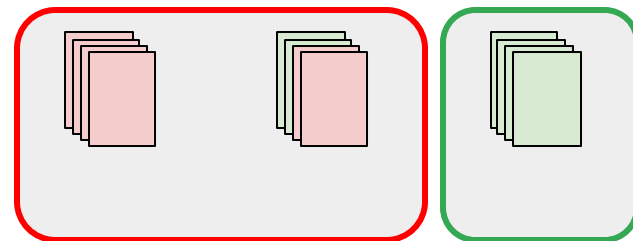


Problem Analysis

Test Outcomes

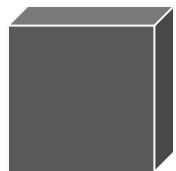


Binary Classification

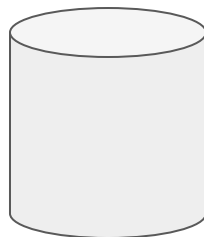


Problem Analysis

Test Outcomes

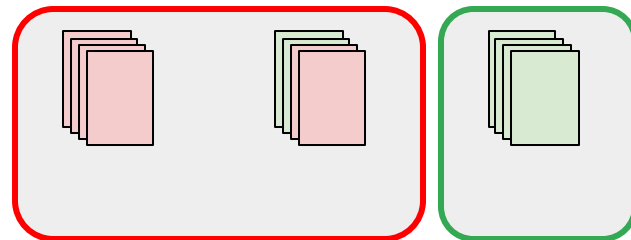


Learning
Algorithm



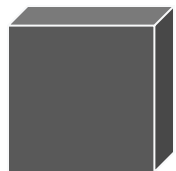
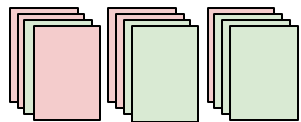
Binary
Classifier

Binary Classification

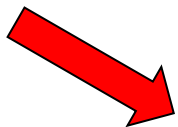


Problem Analysis

Test Outcomes

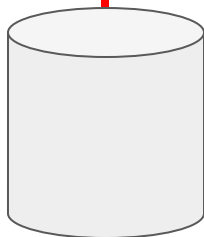


Learning
Algorithm

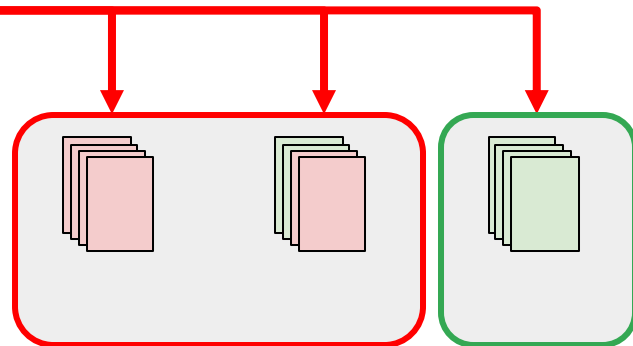


Binary Classification

Classify

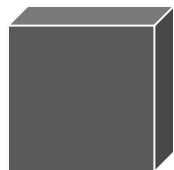


Binary
Classifier

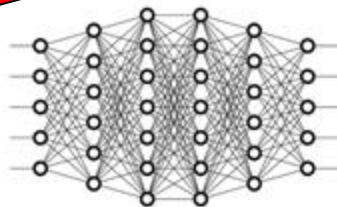
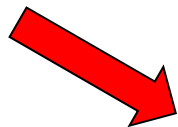


Problem Analysis

Test Outcomes



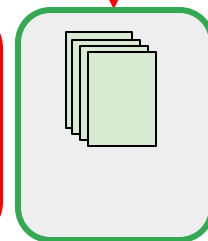
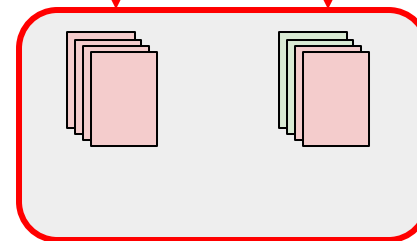
Deep
Learning



Neural
Network

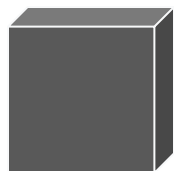
Binary Classification

Classify



Problem Analysis

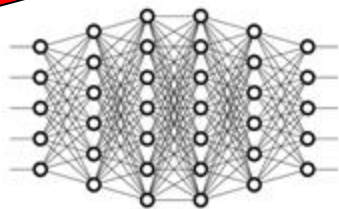
Test Outcomes



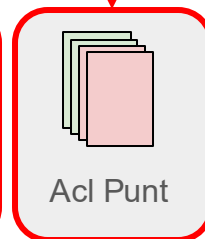
Deep
Learning

Binary Classification

Classify



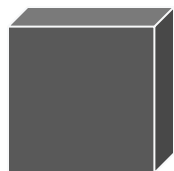
Neural
Network



Explain

Problem Analysis

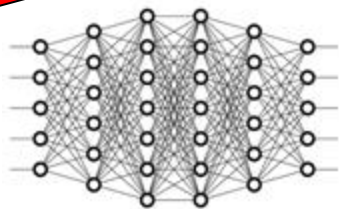
Test Outcomes



Deep Learning

Binary Classification

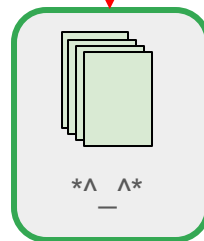
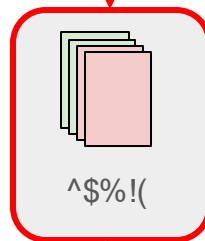
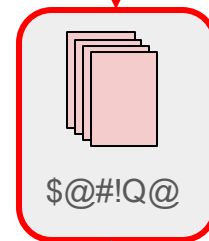
Classify



Neural Network

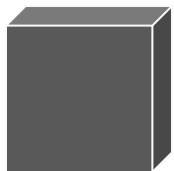
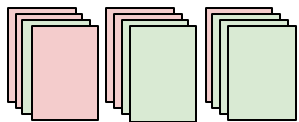


Explain



Problem Analysis

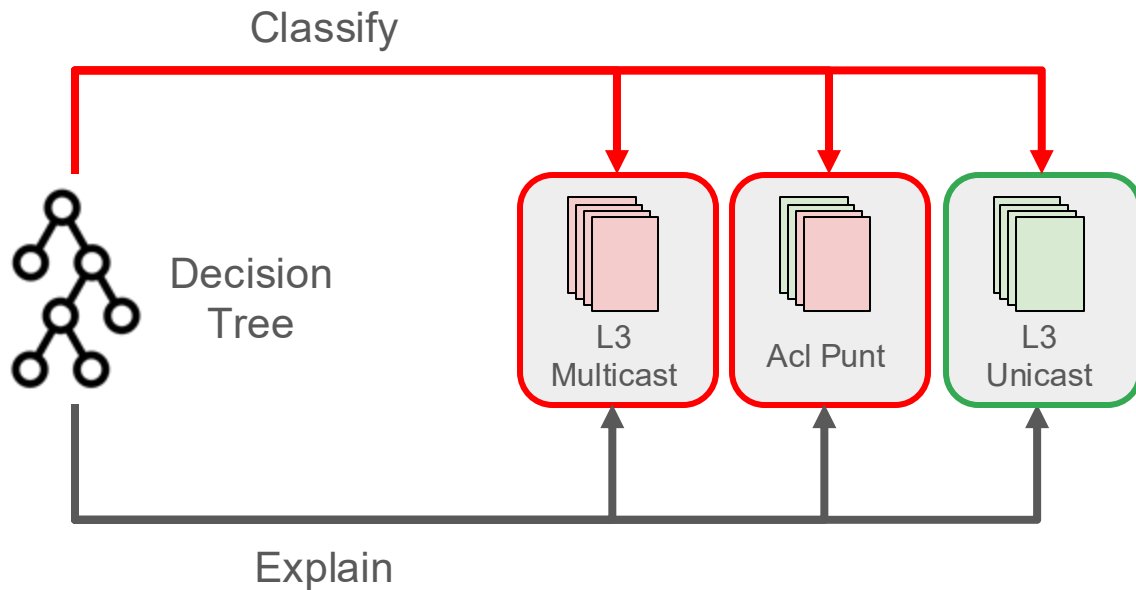
Test Outcomes



Decision Tree
Learning

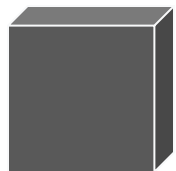
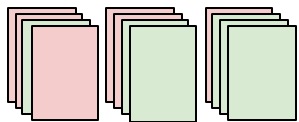


Binary Classification with Interpretable Model



Problem Analysis

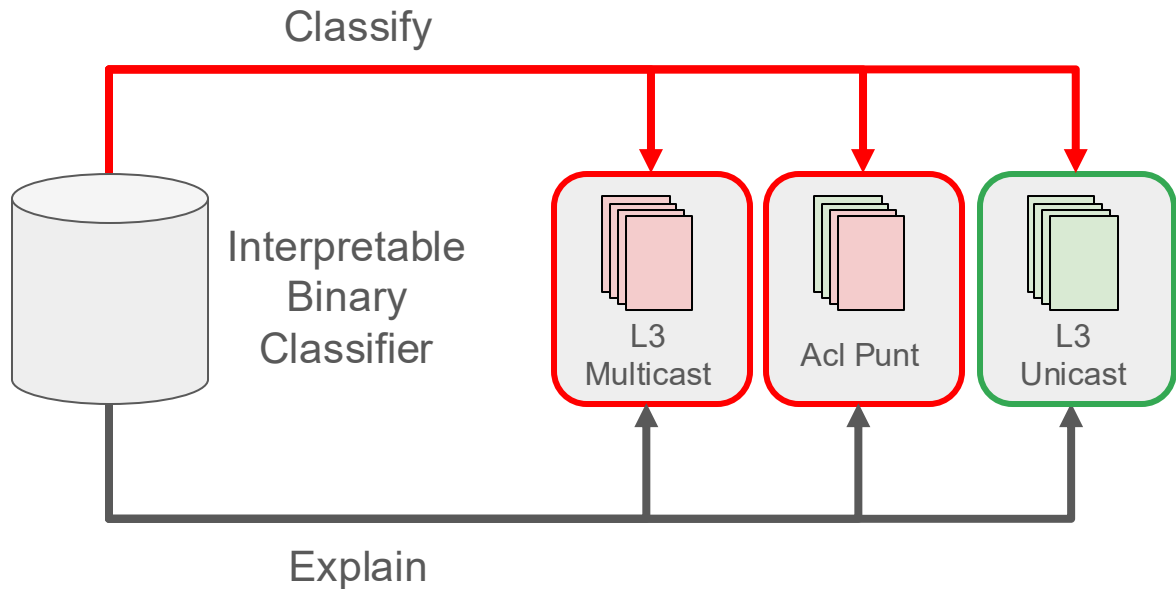
Test Outcomes



Learning
Algorithm

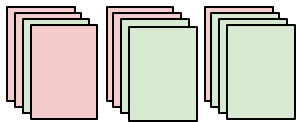
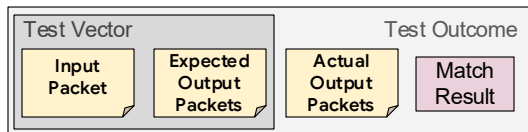


Binary Classification with Interpretable Model

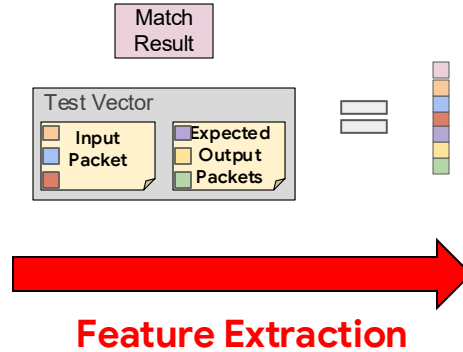
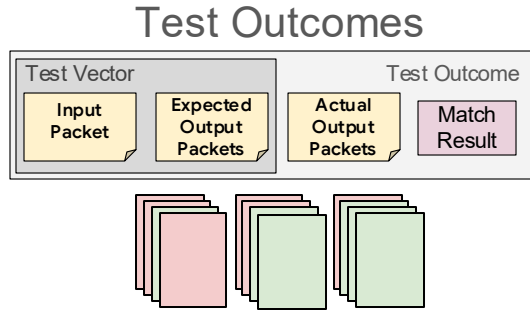


DVaaS Detective: Architecture

Test Outcomes



DVaaS Detective: Architecture



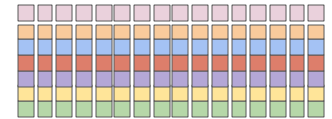
Feature Vectors



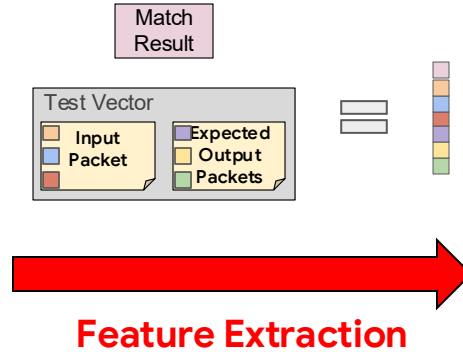
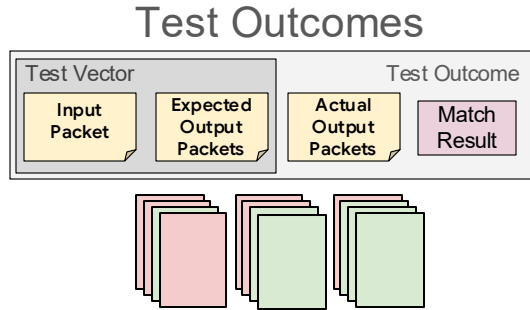
Label: Pass/Fail

Features:

Input EtherType
Input IP Address
Input TTL
Output Packets
Table T entry hit
Action A applied
Many more...



DVaaS Detective: Architecture



Decision Tree



Binary classification
(powered by)



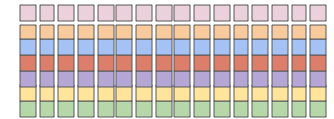
Feature Vectors



Label: Pass/Fail

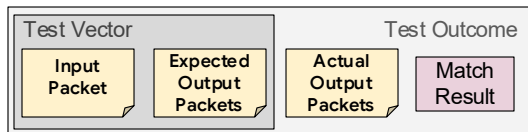
Features:

Input EtherType
Input IP Address
Input TTL
Output Packets
Table T entry hit
Action A applied
Many more...

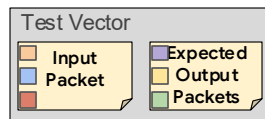


DVaaS Detective: Architecture

Test Outcomes



Match
Result



Feature Extraction

Feature Vectors



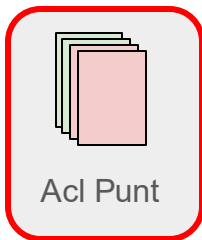
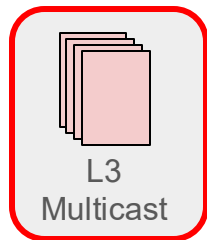
Label: Pass/Fail

Features:

Input EtherType
Input IP Address
Input TTL
Output Packets
Table T entry hit
Action A applied
Many more...



Explanations

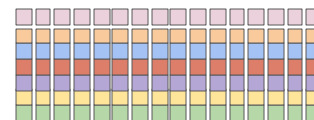


Decision Tree



Parse and
Interpret
The Tree

Binary classification
(powered by)



DVaaS Detective Design Principles

- Synthesize information-dense signal from large amount of noisy data

DVaaS Detective Design Principles

- Synthesize information-dense signal from large amount of noisy data
- Maximize signal and minimize noise $\Rightarrow$ Large signal-to-noise ratio
 - Noise: Amount of content outputted
 - Signal: Useful Information received by user

DVaaS Detective Design Principles

- Synthesize information-dense signal from large amount of noisy data
- Maximize signal and minimize noise $\Rightarrow$ Large signal-to-noise ratio
 - Noise: Amount of content outputted
 - Signal: Useful Information received by user
- Treat user attention as a precious and highly-limited resource

Agenda

Debugging Pain

Background and Problem with DVaaS

Solution: DVaaS Detective

Status & Future Work

Status

- We are gaining experience with the technique at google :)
- Significantly helped explain issues (avoided extensive labor)
- Quality benchmarking with case studies guide development

Status

- We are gaining experience with the technique at google :)
- Significantly helped explain issues (avoided extensive labor)
- Quality benchmarking with case studies guide development

Future Work

- More rigorous feature engineering (empirical right now)
- Identifying and providing example test outcomes
- LLM and Agentic AI

